APPENDIX XII

Instruction formats

```
i = instruction opcode
a = register (r0-3,r13-16)
b = register (r0-3,r13-16)
c = register (r0-3,r13-16)
h = register high (r0-r31)
q = condition code
u = unsigned immediate
s = signed immediate
```

Op	Format	Instruction	Operands	Comment
0	iiiiqqssssssss	bal/beq/bne	s8	; if cc pc=(pc&0xfffffffc)+(s8<<1)
	iiiii1qqssssss	bgt/bge/blt/ble	s6	; if cc pc=(pc&0xfffffffc)+(s6<<1)
1	iiiissssssssss	bl	s10	; blink=pc; pc=(pc&0xfffffffc)+(s10<<2)
2	iiiiaaabbbiiii	op	a,a,b	; op = sub/and/or/xor/asl/asr/lsr/
			a,b,1	; asl1/asr1/
			a,b,2	; asl2/asr2/
			0,a,b	; and.f/mul64/?/?/s_op
	iiiiaaaiii1111	s_op	a,a	; s_op=extb/extw/sexb/sexw/
			[a]	; j/jl/
			a,a,a	; sub.ne/i_op
	iiiiiii1111111	i_op3		; i_op=brk/j [blink]/st blink[sp,4]/
3	iiiiaaaiuuuuuu	mov/cmp	a,u6	
4	iiiiaaaiiuuuuu	add/sub/?/?	a,a,u5	
5	iiiiaaahhh00hh	mov	a,h	
	iiiiaaahhh01hh	add	a,a,h	
	iiiiaaahhh1ihh	mov/cmp	h,a	
6	iiiiaaa000iuuu	ld/st	a,[fp, -u3]	; a=mem[fp - (u3 << 2)]
	iiiiaaa001iuuu	add/?	a,[fp, -u3]	; a=fp - (u3 << 2)
	iiiiaaaiiuuuuu	asl/asr/lsr	a,a,u5	
7	iiiiaaabbbuuuu	ld	a,[b,u4]	; a=mem[b + (u4<<2)]
8	iiiiaaabbbuuuu	ldb	a,[b,u4]	; a=mem[b + u4]
9	iiiiaaabbbuuuu	ldw	a,[b,u4]	; a=mem[b + (u4<<1)]
A	iiiiaaabbbuuuu	st	a,[b,u4]	; a=mem[b + (u4<<2)]
B	iiiiaaabbb0uuu	stb	a,[b,u3]	; a=mem[b + u3]
	iiiiaaabbb1uuu	stw	a,[b,u3]	; a=mem[b + (u3<<1)]
C	iiiiaaabbbiuuu	add/sub	a,b,u3	; a=b op u3
D	iiiiaaauuuuuuu	ld	a,[pc,u7]	; a=mem[(pc&0xfffffffc)+(u7 << 2)]
E	iiiiaaauuuuuuu	ld	a,[gp,u7]	; a=mem[gp + (u7 << 2)]
F	iiiixxxxxxxxxx	reserved		

Other possible formats:

iiiiaaabbb0ccc	ld	a,[b, c]	; a=mem[b + c]
iiiiaaabbb1ccc	add	a,b,c	; a=b+c
iiiiaaauuuuuuu	add	a,pc,u7	; a=(pc&0xfffffffc)+ (u7 << 2)